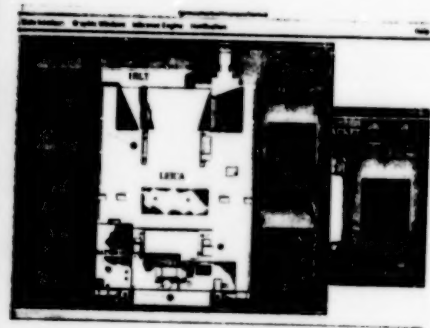
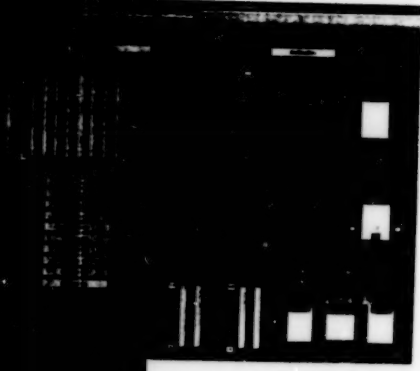
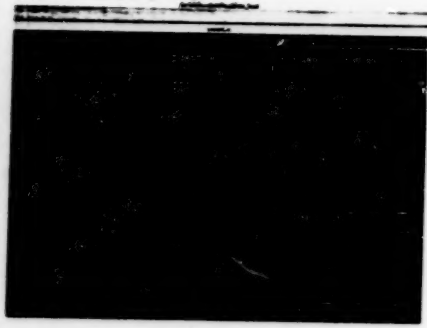


0820-I

NAS 1.83.SP 1

GENERIC SPACECRAFT ANALYST
ASSISTANT (GENSAA) AND
THE GENERIC INFERENCEAL
EXECUTOR (GENIE)

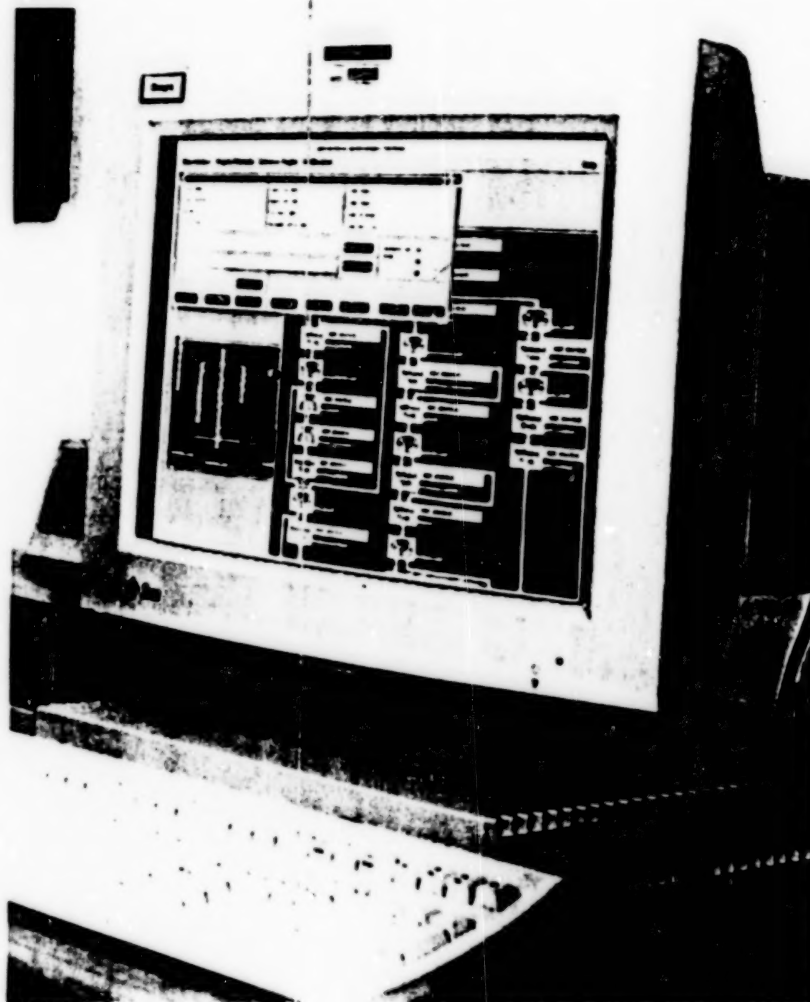


97-022381/CS

C

GenSAA Overview

For nearly 25 years, spacecraft missions were operated in essentially the same manner: human operators monitored displays filled with alphanumeric text watching for limit violations or other indicators that signaled problems. Only in recent years have graphical user interfaces and expert systems been accepted within the control center environment to help reduce operator workloads. Unfortunately, the development of these systems is often time consuming and costly. At the NASA Goddard Space Flight Center (GSFC), a domain-specific expert system development tool called GenSAA makes development quick and easy. Through the use of a highly graphical user interface and point-and-click operation, GenSAA facilitates the rapid, "programming-free" construction of intelligent graphical monitoring systems to serve as real-time, fault-isolation assistants for spacecraft analysts. Although specifically developed to support real-time satellite monitoring, GenSAA can support the development of intelligent graphical monitoring systems in a variety of space and commercial applications.



"GenSAA has demonstrably improved operations efficiency and increased capability through automation of routine operations functions and introduction of effective visualization capability within a control center. It is also a pathfinder and enabler..."

*Chief, Mission Operations and
System Development Division
NASA's Goddard Space Flight Center*

"In a time when operations budgets are being cut back, GenSAA/Genie offers solutions for meeting lower staffing profiles while maintaining quality of service to our customers."

*Systems Engineer, AlliedSignal
Technical Services Corporation*

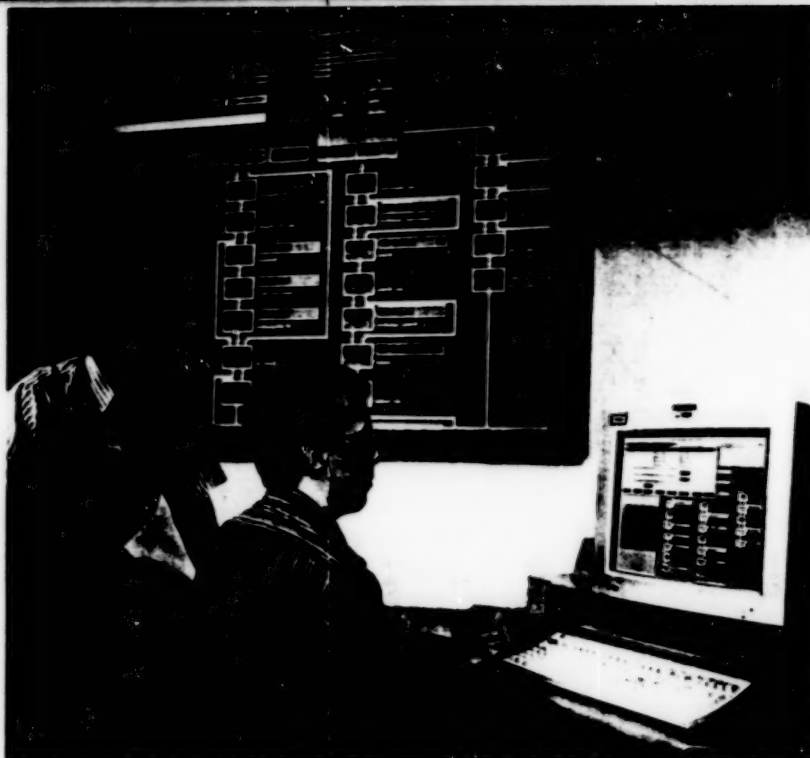
I

GenSAA/Genie at a Glance

An Expert System Development Tool

GenSAA is a UNIX-based software tool that enables spacecraft analysts to rapidly build simple, yet highly graphical, expert systems that are capable of performing real-time spacecraft monitoring, fault detection, and fault isolation functions. Genie is an extension of GenSAA that adds spacecraft commanding and spacecraft pass automation functions to GenSAA. GenSAA/Genie was created as an alternative to more complex, expensive, commercially-available expert system development environments. A primary goal in the development of GenSAA was to create a tool such that it was easy for domain experts (e.g., a spacecraft engineer) to transfer their knowledge directly into a GenSAA expert system.

GSFC wanted to take advantage of the potential of graphical expert systems but needed a way to improve productivity in developing and maintaining these systems. GenSAA/Genie addresses these objectives by insulating the expert system developer from the programming details by way of a "direct-manipulation environment" in developing intelligent graphical monitoring systems.



Characteristics of an Expert System Designed with GenSAA/Genie

Easily Created and Modified - The process for developing specific expert systems using GenSAA is straightforward and it can be performed by trained users, e.g., spacecraft analysts. No programming languages need to be learned. No compilation or link step is necessary before executing GenSAA expert systems, thus changes can be made dynamically.

Rule-based - GenSAA/Genie supports the use of rules to represent spacecraft and payload fault detection, monitoring, and fault isolation knowledge. Rule-based representations are intuitive and can closely model human thoughts used to describe many of the reasoning processes.

Graphical - The GenSAA operational user interface is X-Windows-based. It supports both textual and graphical presentations of health and status information as well as expert system fault isolation conclusions. Hyperlink techniques are supported to simplify navigation between GenSAA windows. Genie provides a graphical pass script builder to automate spacecraft pass operations.

Transparently Interfaced with the Data Source - GenSAA/Genie is being used to create expert systems that support analysts in spacecraft control centers that use the Transportable Payload Operations Control Center (TPOCC) architecture. TPOCC is a UNIX-based control center system architecture that is being used for several spacecraft missions such as Solar Anomalous and Mesospheric Particle Explorer (SAMPEX), X-Ray Timing Explorer (XTE), Interplanetary Physics Laboratory (WIND), Polar Plasma Laboratory (POLAR), and Solar Heliospheric Observatory (SOHO). GenSAA understands TPOCC data formats, providing transparent access to the data. GenSAA can also be configured to transparently receive data from sources other than TPOCC.

Real Time - GenSAA/Genie expert systems are driven by real-time data sources and inferences are made in near-real-time spacecraft telemetry that indicate the current status of the spacecraft and its operation. Increased productivity is just one of the advantages offered by GenSAA. GenSAA's powerful set of user interface development and management tools allows an organization to create a consistent look and feel for the user interfaces of all its expert systems, reducing user training time, while heightening efficiency. In addition, GenSAA isolates the user interface code from the application code, so user interface changes can be made quickly and conveniently.

above right: Genie autonomously takes a spacecraft pass

Productivity is Just the Beginning

GenSAA's Architecture

GenSAA is composed of an integrated set of utilities, called the GenSAA Workbench, and a runtime executive called the GenSAA Runtime Framework. The expert system developer uses the Workbench off-line for laying out the graphical user interface, defining expert system fault detection rules, and selecting the telemetry data to drive the system. GenSAA insulates the spacecraft analyst from the complicated programming details of the data source (spacecraft ground system) and the graphical user interface. The Workbench creates a set of files that are used by the GenSAA Runtime Framework to define the executable GenSAA expert system.

The GenSAA Workbench is an off-line tool for specifying:

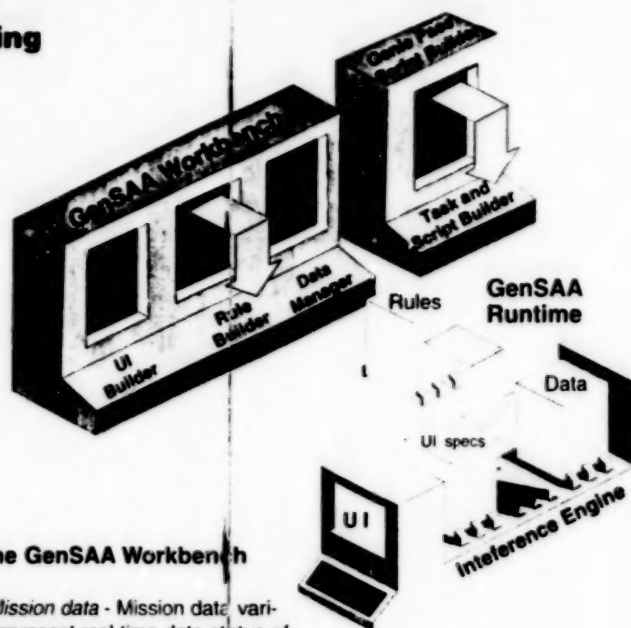
- Telemetry data
- Expert system rules
- User interface layout

The Runtime Framework is a runtime executive that:

- Provides specified connections to the data source(s)
- Provides specified connections to the inference engine
- Provides real time inferencing
- Executes defined applications created with the GenSAA Workbench
- Displays specified user specific interface

The GenSAA Workbench operates in an off-line mode (i.e., without being connected to a data source) on a Unix system and is created with three integrated utilities. The three utilities are: the Data Manager, the Rule Builder, and the User Interface Builder.

The Data Manager – is used to construct and edit the four types of variables that are received from external data sources, exchanged with other GenSAA expert systems, or associated with GenSAA's Graphic Objects.

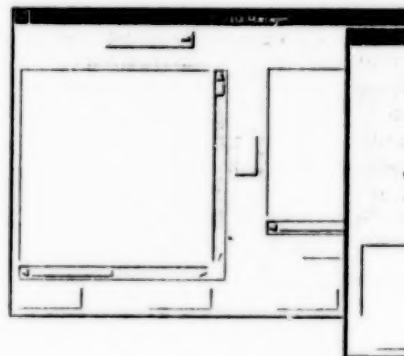


The GenSAA Workbench

- (1) **Mission data** - Mission data variables represent real-time data status of the monitored data (e.g., spacecraft ground system data) and related ground support systems. This data originates from the external data source telemetry data server.
- (2) **User-defined data** - User-defined data variables represent expected operating modes and equipment configurations. This data originates from the user input.
- (3) **Inferred data** - Inferred data variables represent conclusions inferred by rules in the rule base. This data originates from the inference engine.
- (4) **EKG data** - Externally-generated GenSAA (EGG) variables are data available from other GenSAA expert systems that run concurrently. This data originates from other expert systems.

The Rule Builder – is used to create the rule base for a GenSAA expert system. The rule base is a set of expert system rules in "condition-action" (or "if-then") format that may infer new facts based on current facts. A rule contains one or more conditions and one or more actions. GenSAA uses the C Language Integrated Production System (CLIPS) inference engine and rule syntax. The GenSAA Rule Builder can automate the creation of the CLIPS rule syntax using graphical means. The Rule Builder includes the Rule Editor window, the Condition Builder window, and the Action Builder window.

above: The GenSAA Architecture
below: The GenSAA Data Manager and Data Details Window



Using GenSAA

The GenSAA Rule Editor, Condition Builder, and Action Builder

The User Interface Builder – is used for creating graphical user interfaces for GenSAA expert systems. Users can populate a user interface with primitive objects (e.g., text and polygons) or with complex objects (e.g., dials, bars, and xy plots). The user interface of a GenSAA expert system consists of a workspace, graphic windows, and graphic objects. These graphic elements can be dynamically created, customized, and linked with data without programming by using mechanisms provided in the User Interface Builder.

The GenSAA Runtime Framework – provides the basic operational environment for a GenSAA expert system. The components of the GenSAA Runtime Framework are reused in each GenSAA expert system. They control the operation of a GenSAA expert system during execution. They read the data interface specification, rule base, and user interface specification files created by the GenSAA Workbench to determine a specific behavior. The GenSAA Runtime Framework is implemented as a pair of UNIX processes that communicate with one another via sockets. A third process is typically an external data source (e.g., control center data server). Their functions are:

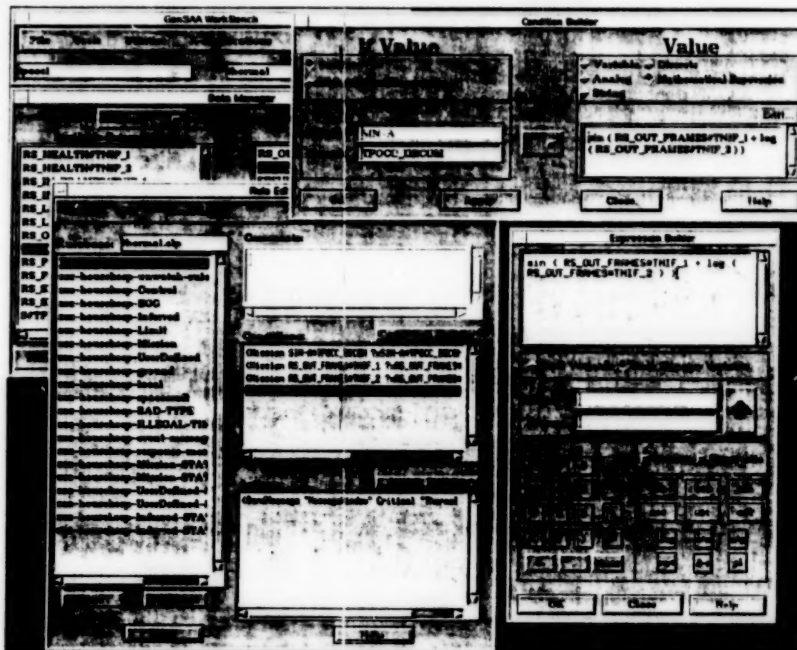
User Interface Process – manages the operation of the expert system's user interface. It handles the user input, manages the display of windows (that may contain both text and graphics), and updates the visual attributes of the graphical objects. The User Interface Process also handles data requests and distribution. The user interface windows, data-driven objects, and interaction objects are created in the GenSAA User Interface Builder.

Inference Engine Process – manages the firing of rules in the rule base. A rule is fired when all of its conditions are satisfied; the conditions will often involve the current values of mission, user-defined, EGG, and inferred data variables.

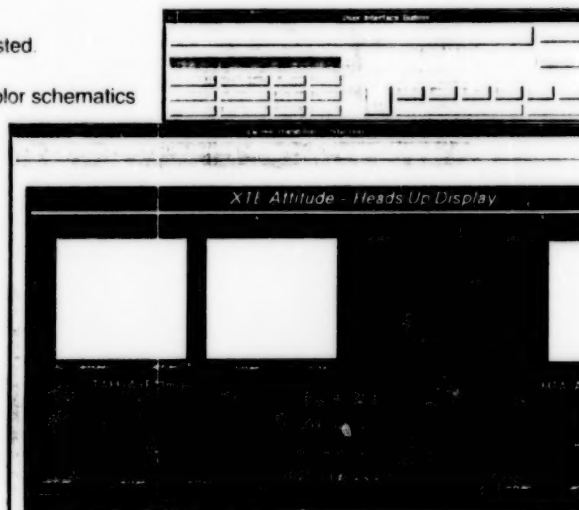
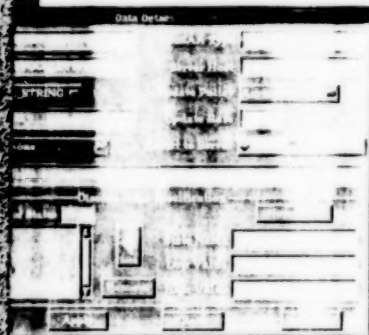
External Data Server – sends data to the expert system as requested.

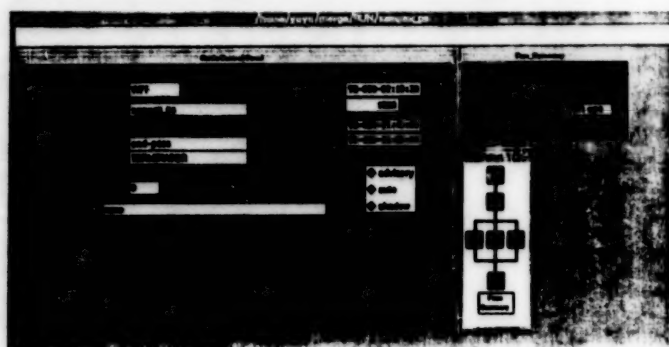
The user interface of a GenSAA expert system typically includes color schematics and animated data-driven objects (such as rotating meters, sliding bar graphs, and toggle switches) that graphically display the dynamic values of telemetry data, user-defined data, and inferred conditions. The user interfaces also typically contain hyperlinks that

make it easier for the spacecraft analyst to quickly display other graphics windows.



above right: The Condition Builder
below right: User Interface Builder





Commanding the Spacecraft with the Generic Inferential Executor (Genie)

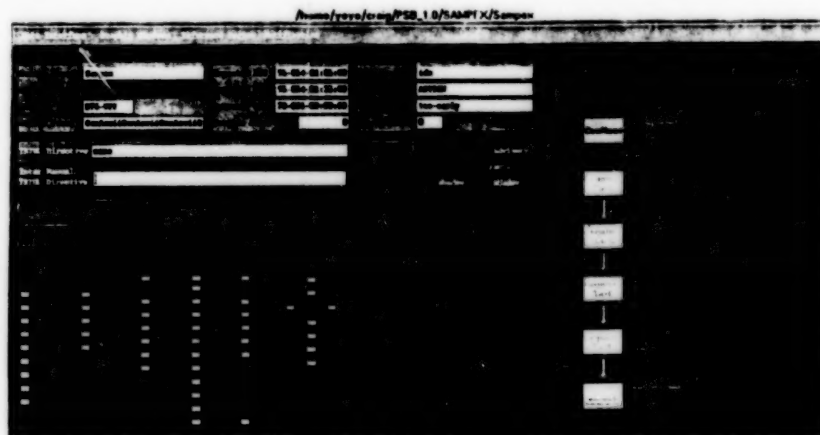
Most spacecraft-ground contacts, or passes, are routine activities, during which the Flight Operations Team (FOT) oversees the downloading of tables and science data, as well as the loading of onboard stored commands. The process is well understood and repetitive, in other words, an ideal automation candidate. A FOT can use automation to allow them to do more with constant or decreasing resources. One ideal application is multi-spacecraft control centers, where operation of mature spacecraft can be largely automated, allowing attention to be focused on newly launched satellites. The FOT is immediately contacted if the automation system detects a non-nominal condition, making this approach low risk.

Genie is a tool that allows easy construction of pass automation applications. A pass is a short period of time (e.g., 10 minutes at 90-minute intervals) during which the control center is in contact with the spacecraft, sending commands and receiving data. A graphical workbench allows the user to build pass-script templates that encode the tasks necessary to mimic the sequence of activities that spacecraft controllers typically perform during a pass. These templates are then configured by a mission planner with data specific to a particular pass. The script is loaded into the Genie runtime system during pre-pass preparations and executed. A graphical status and control display tracks progress during the pass. Genie runs in the following three modes:

- **Shadow Mode** - tells what it would have done and what kind of commands it would have sent in any kind of situation.
- **Advisory Mode** - makes recommendations as to what to do and asks the operator whether or not to perform the command.
- **Full Automation Mode** - commands the spacecraft and tells the operator what command it has performed.

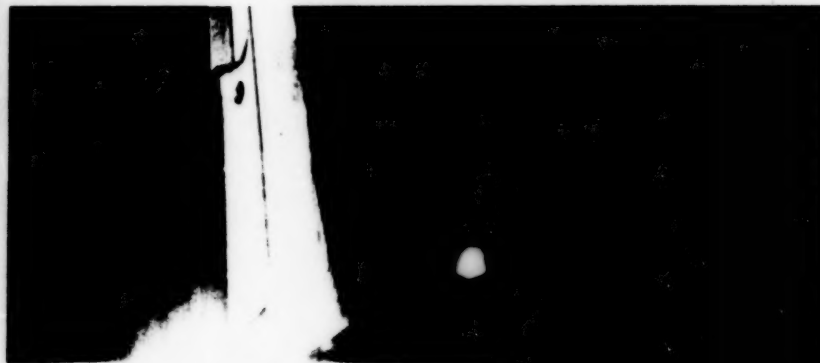
Genie has been used successfully with the SAMPEX spacecraft in all three of the above modes.

Development of Genie required three efforts: 1) minor modifications were made to GenSAA software, adding interfaces to receive events from and send commands to the control center software; 2) a graphical pass script builder was implemented; and 3) a script executor was "coded" in rules for the C Language Integrated Production System (CLIPS), the inference engine at the heart of both GenSAA and Genie. Applications for new missions are straightforward to build once routine spacecraft operation is established.



above left and below right: Genie User Interface

Interfacing to External Sources



Co-operating GenSAA Expert Systems

GenSAA expert systems are intended to be relatively simple, modular expert systems with small rule bases. A typical GenSAA expert system might monitor and isolate faults for one spacecraft subsystem. To handle more complex monitoring situations, GenSAA modular expert systems can be built for a discrete subsystem or function and can execute concurrently and can share key conclusions with one another using a "publish-and-subscribe" model of communication. This way system level expert systems can simply monitor the conclusions made by the subsystem level expert systems. To perform the publish-and-subscribe method of information sharing, a third-fourth component of the GenSAA Runtime Framework, the GenSAA Data Server, is used as a central repository in which GenSAA expert systems can "publish" information and from which other "subscribing" GenSAA expert systems can receive the information when published. Additionally, external applications can "publish and subscribe" to this data. An alternative method for sharing data between GenSAA expert systems and external data sources is via the CLIPS-socket interface. This interface allows for expert system rules to send data to an external source and the external source can send data back into the inference engine.

GenSAA's Implementation and Operational Environment

GenSAA is a client/server system using the X-Window System with the Motif library and Window Manager for a graphical operator interface.

GenSAA is designed to operate in a network of computers. It consists of a small set of specialized front-end processors and UNIX-based workstations on an Ethernet network using the TCP/IP network protocol. GenSAA can operate without the TPOCC.

Benefits of GenSAA/Genie

Productivity

GenSAA assists the controllers with data monitoring resulting in more efficient and effective operations. Genie provides automated commanding.

Efficiency

GenSAA enables rapid iterative development of expert systems within the application domain.

Cost Savings

GenSAA expert systems reduce development costs, improve operations efficiency, and keep expertise in-house as personnel turnover occurs. Genie reduces staff required for routine operations by completing automating spacecraft passes.

Flexibility

User interface changes can be made via the Workbench alone, with no changes to the source code.

Ease of Use

The GenSAA WorkBench allows non-programmers to create sophisticated user interfaces easily with the drag and drop technique.

Reusability

User interface designers can reuse and share customized interaction objects, entire windows, and even sets of related windows, avoiding unnecessary duplication of effort. Expert system rules can also be shared.

Timely

Reduces development time and effort; allows quick and accurate response to necessary modification.

Training Tool

Hands-on environment for training spacecraft controllers in a safe, off-line mode.

Potential Commercial Uses

GenSAA could be used for any routinely monitored data source where anomaly detection, isolation, and correction are required including:

- Spacecraft, airplane, and automobile operations
- ⚙ Industrial process control
- ⚡ Power plant operations
- 🔒 Security systems
- 🏥 Biomedical monitoring of life support and diagnostic equipment

BEST AVAILABLE COPY

Conclusion

The NASA Software Advisory Council has chosen GenSAA for Honorable Mention in the 1996 NASA Software of the Year competition. GenSAA was selected for its high degree of innovation and ease of use, as well as its importance to NASA's mission.

At the GSFC, there is a strong interest in reducing satellite operations costs with modern automation techniques. GenSAA/Genie was developed to allow highly graphical expert systems to be created quickly and in a cost-effective manner. GenSAA/Genie is well suited for use on spacecraft projects that involve a series of similar, but non-identical, missions such as NASA's Small Explorer (SMEX) and International Solar-Terrestrial Physics (ISTP) programs. It is currently being applied to the SAMPEX, FAST, XTE, WIND/POLAR, and the SOHO missions. GenSAA/Genie expert systems that are under development or include support for are THMM, FAST, SWAS, ACE, GRO, EUVE, and NOAA spacecraft. The EDOS and MIDEX missions are currently evaluating GenSAA/Genie for use. GenSAA provides an evolutionary approach to automating satellite operations from simple monitoring functions to full pass automations. Although GenSAA is expected to increase automation and operator effectiveness, it has not been determined whether it will scale up to the larger problem of a fully automated control center.

GenSAA
patent pending

NP-1997(04)-012-GSFC

6

Contact

END

02-23-00